

REKAYASA PERANGKAT LUNAK BERBASIS SUMBER TERBUKA UNTUK MEMBANTU PELAKSANAAN AUDIT SISTEM INFORMASI

Hari Setiabudi Husni

Jurusan Komputerisasi Akuntansi, Fakultas Ilmu Komputer, Universitas Bina Nusantara,
Jln. K.H. Syahdan No. 9, Kemanggis/Palmerah, Jakarta Barat 11480
hari.setiabudi@binus.ac.id

ABSTRACT

This research was conducted on one budget period in 2009 funded by DIKTI young lecturer research project grant. The main research location is Bina Nusantara University. Due to tight research schedule, it was necessary to take some strategic steps to fulfill research goals. One of the strategic steps was to invite several experts in software industry to give advices regarding open source software engineering issues. The first achievement was findings of some open source software that could assist on auditing information systems. Afterwards, comparison from technical and functional aspects resulted in the best software to be tested for implementation and usage, namely ZenossCore. The final output of this research is successful reengineering of the source code for virtual file testing.

Keywords: open source software, audit implementation, information system

ABSTRAK

Penelitian ini dilaksanakan selama satu periode anggaran proyek penelitian dosen muda DIKTI tahun 2009. Lokasi penelitian berada di Universitas Bina Nusantara. Karena sempitnya waktu yang tersedia, maka untuk mempercepat pencapaian tujuan-tujuan penelitian, diterapkan beberapa langkah strategis. Salah satu langkah strategis tersebut adalah mengundang beberapa ahli perangkat lunak untuk diajak berkonsultasi mengenai cara melakukan rekayasa yang cepat dan tepat terhadap perangkat lunak yang ada. Hasil penelitian yang dicapai pertama adalah berhasil ditemukannya beberapa perangkat lunak berbasis sumber terbuka untuk membantu pelaksanaan audit sistem informasi. Selanjutnya, dilakukan perbandingan dari aspek teknis dan fungsional yang menghasilkan satu saja perangkat lunak terbaik untuk dilakukan uji coba implementasi dan pemanfaatan yakni ZenossCore. Hasil akhir dari penelitian ini adalah berhasilnya rekayasa sumber kode (source code) perangkat lunak tersebut untuk diuji coba file virtualnya.

Kata kunci: perangkat lunak berbasis sumber terbuka, pelaksanaan audit, sistem informasi

PENDAHULUAN

Indonesia sebagai sebuah negara berkembang memiliki permasalahan kesenjangan teknologi dengan negara maju. Hal ini menghambat sumber daya manusia untuk penguasaan teknologi apabila tidak disiasati dengan baik. Salah satu aspek yang menyulitkan dalam upaya mempersempit kesenjangan adalah mahalnya produk-produk aplikasi berlisensi ketika masuk Indonesia. Namun, dengan adanya produk aplikasi dengan sumber terbuka (*open source*), diharapkan dapat memberikan solusi permasalahan ini dalam jangka pendek maupun jangka panjang.

Saat ini komunitas pendukung pemanfaatan aplikasi sumber terbuka telah sangat banyak variasinya, terentang dari komunitas untuk aplikasi sistem operasi hingga aplikasi untuk manajemen warung internet. Keberadaan komunitas ini di Indonesia selain menunjukkan adanya kegairahan dalam mengembangkan intelektualitas berkualitas secara legal, juga memperlihatkan bahwa sumber daya manusia Indonesia tidak kalah dalam persaingan penguasaan teknologi perangkat lunak.

Tidak jauh berbeda dengan kondisi pemanfaatan aplikasi berlisensi pada bidang lain di Indonesia, untuk teknik audit dengan memanfaatkan komputer (CAATs - *Computer Assisted Audit Techniques*) terhambat perkembangannya

karena sulitnya mendapatkan aplikasi yang murah dan memiliki fungsi lengkap. Untuk itu dibutuhkan penelitian yang sistematis dan terstruktur untuk menjawab kebutuhan akan perangkat lunak berbasis sumber terbuka, untuk membantu pelaksanaan audit sistem informasi.

Teknik audit dengan bantuan komputer adalah cara di mana pemanfaatan aplikasi komputer dilakukan untuk mengumpulkan dan membantu mengumpulkan data dari bukti-bukti audit. Setiap teknik memiliki fungsi yang berbeda, contoh (1) Perangkat lunak komersil seperti Microsoft Excel, Word, dan lainnya digunakan oleh auditor untuk analisis data dari file klien, menulis program audit, dan lainnya; (2) Perangkat lunak audit umum memiliki berbagai bentuk, bisa berbentuk perangkat lunak komersil atau pengembangan firma audit. Tujuan perangkat ini adalah untuk interogasi, ekstraksi, dan kadang menganalisis informasi dari komputer manajemen. Aplikasi perangkat lunak ini bisa digunakan untuk pengujian kendali atau pada prosedur substantive; (3) Simulasi parallel. Prosedur ini memberikan bukti efektifitas prosedur kendali desain pemrograman juga efektifitas operasinya; (4) Analisis kode program dilakukan pada kode program klien untuk memastikan instruksi yang dituliskan sesuai dengan dokumentasi program yang sebelumnya dievaluasi oleh auditor; dan (5) Uji data adalah prosedur yang memanfaatkan data uji untuk melihat proses aplikasi pada sistem informasi

klien, tanpa mengikutsertakan data sebenarnya (*Activity Based Risk Evaluation Model of Auditing*, 2008).

Perangkat lunak berlisensi diproduksi dan dijual oleh lembaga komersil yang berinisiatif sangat kuat dalam mengiklankan produknya dan menginformasikan kepada konsumen potensial tentang keunggulan paket mereka. Perangkat lunak berlisensi biasanya disediakan oleh sebuah lembaga komersil (Comino and Manenti, 2003).

Pada awal tahun 1980, Microsoft mulai menulis perangkat lunak untuk tujuan utama keuntungan. Yang diberikan hanya *compiled code*, sedangkan *source code* disembunyikan dari pengguna. Aktifitas ini memberikan dampak besar dan dapat disebut sebagai kelahiran perangkat lunak berbasis terbuka. Richard Stallman, peneliti MIT, memulai *Free Software Foundation* (FSF) untuk mengembangkan dan mendistribusi perangkat lunak di bawah *General Public License* (GPL), sementara Bruce Perens memberikan garis besar kriteria dari sebuah perangkat lunak, dan menyebutnya *Open Source Initiative* (OSI) (Potdar and Chang, 2004).

Pada dasarnya, istilah berbasis sumber terbuka adalah perangkat lunak yang *source code* nya terbuka dan tersedia. *Source code* adalah program yang berisi kode asli perangkat lunaknya. Perangkat lunak disebut sumber terbuka jika *source code* mereka bisa dilihat, dibaca, dan ditulis ulang atau modifikasi oleh semua orang. Ketersediaan maksudnya adalah setiap orang bisa mendapatkan kode program secara bebas atau biaya minim (biasanya biaya dari media dan pengiriman atau biaya hubungan *online*) (Comino and Manenti, 2003).

Dibandingkan dengan perangkat lunak dengan lisensi, perangkat lunak berbasis sumber terbuka kurang berkembang dan jarang digunakan secara luas disebabkan tidak mendapat dukungan yang solid dari para pengambil kebijakan, baik

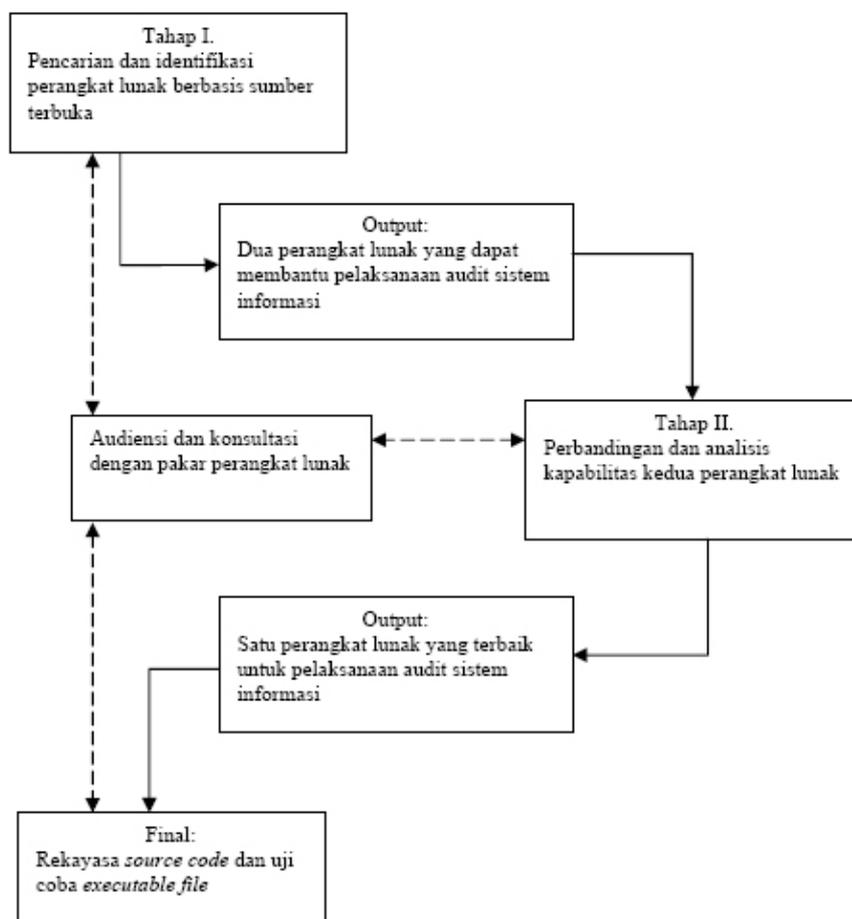
skala nasional maupun lokal. Hal ini menyebabkan sulit berkembangnya pengetahuan teknologi pada generasi muda khususnya dalam wacana persaingan kompetensi penguasaan dan pengembangan teknologi terbaru.

Khususnya pada perangkat lunak yang digunakan untuk membantu audit sistem informasi (CAATs), memang sudah ada yang mencoba mengembangkan serangkaian fungsional audit dalam satu aplikasi seperti yang dapat diunduh dari situs *Sourceforge*. Namun, disebabkan sifat audit yang memiliki keunikan proses pada setiap kawasan dan hambatan bahasa, maka dibutuhkan serangkaian rekayasa dari perangkat lunak tersebut agar siap dan layak digunakan di Indonesia. Di sini keunggulan dari aplikasi perangkat lunak berbasis sumber terbuka, yakni pengguna dari manapun tidak diwajibkan membayar royalti kepada pengembang pertama dari sebuah perangkat lunak. Selain itu, pengguna dapat merekayasa sistem aplikasi secara bebas dan berhak melakukan kustomisasi yang dibutuhkan agar aplikasi tersebut dapat digunakan secara mandiri.

Meskipun memiliki kemudahan, aplikasi berbasis sumber terbuka tetap membutuhkan kerja keras, konsistensi, komitmen, dan manajemen kerja yang baik agar terwujud sebuah aplikasi yang mandiri dan lengkap secara fungsional. Untuk itu dibutuhkan sebuah program pengembangan dan rekayasa yang menyeluruh agar dapat menjawab permasalahan ini sehingga tujuan penelitian ini, yakni merekayasa sebuah purwa rupa perangkat lunak berbasis terbuka yang dapat digunakan dapat memberikan manfaat untuk membantu pelaksanaan audit sistem informasi di Indonesia.

METODE PENELITIAN

Metode penelitian yang digunakan dalam penelitian ini adalah:



Gambar 1 Alur Metode Penelitian

HASIL DAN PEMBAHASAN

Pada tahap pertama penelitian, diperoleh 2 jenis perangkat lunak. Keduanya didapat setelah melakukan serangkaian penelusuran literatur dan halaman web, terutama yang terkait dengan topik pemanfaatan perangkat lunak berbasis sumber terbuka untuk membantu pelaksanaan audit sistem informasi. Untuk perangkat lunak yang pertama adalah *aperture-1.2.0*, didapat dari situs *sourceforge.net* dan untuk yang kedua adalah *ZenossCore 1.1.1-x86*, yang juga didapat dari situs yang sama.

Keduanya memiliki lisensi GNU *General Public License* (GNU GPL atau hanya GPL), yaitu lisensi perangkat lunak gratis yang paling banyak digunakan. Aslinya ditulis oleh Richard Stallman untuk proyek GNU. GPL adalah contoh lisensi penggandaan terkuat yang populer dan dikenal yang menjadikan hasil turunan kerjanya tetap berada pada lisensi yang sama. Dengan filosofi ini, GPL mengizinkan penerima aplikasi perangkat lunak menggunakan hak mendefinisikan ulang aplikasi tersebut dan kebebasan, meskipun aplikasi tersebut diubah dan ditulis ulang.

Aperture adalah kerangka Java untuk mengekstraksi dan *query* konten teks lengkap dan metadata dari berbagai sistem informasi (misalnya file sistem, situs web, mail kotak) dan format file (misalnya dokumen, foto) yang terjadi dalam sistem. *Aperture* dimulai sebagai sebuah kerja sama antara Pusat Penelitian Jerman untuk *Artificial Intelligence* (<http://www.dfki.de>) dan perusahaan perangkat lunak Belanda Aduna (<http://www.aduna-software.com>).

Kedua organisasi telah berbagi perangkat lunak yang dihasilkan masing-masing, dan memiliki karakteristik tertentu seperti penargetan pencarian desktop menggunakan teknologi *Semantic Web*. Oleh karena itu, mereka tentu harus memecahkan masalah teknis yang sama seperti inkremental *crawler* dari sistem file, teks dan ekstraksi metadata dan pengindeksan dan *query* metadata. *Crawler* adalah sebuah program komputer yang melakukan penelusuran informasi memanfaatkan metode tertentu dan berjalan otomatis pada sebuah sistem. Hal ini membuat mereka menyadari bahwa melalui kerjasama, dalam hal ini mereka bisa mendapatkan kode yang lebih baik daripada upaya masing-masing yang hasilnya lebih rendah. Lebih lanjut, hal ini akan memungkinkan orang lain untuk berkontribusi juga.

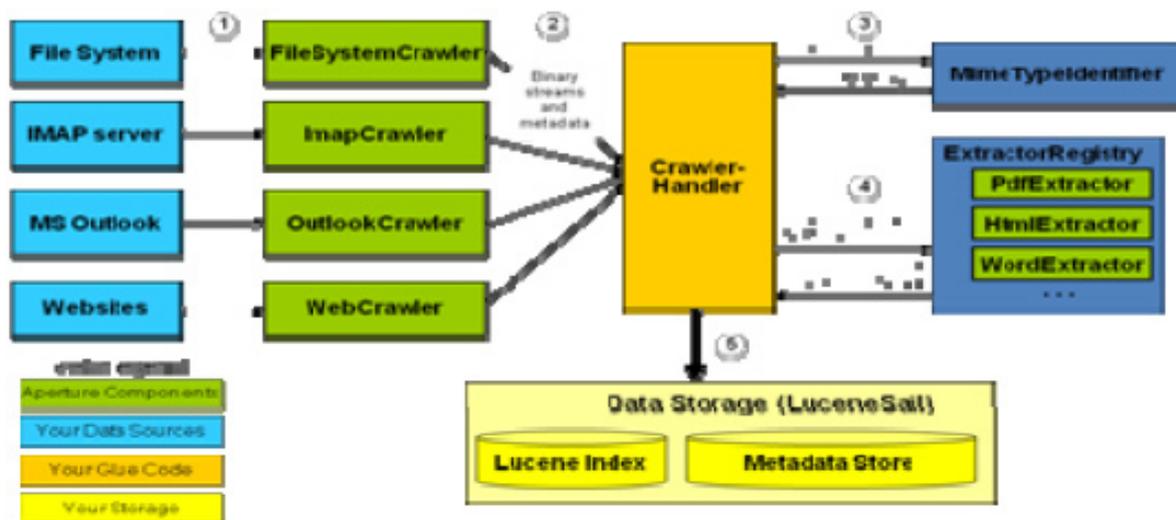
Pada musim panas 2005, mereka memutuskan untuk memulai kerjasama proyek *open source* yang akan mengkombinasikan dengan *crawler* dan kode pengindeksan yang sudah dikembangkan *in-house*, yang akan berfungsi sebagai dasar untuk pengembangan selanjutnya. Secara

teknis, *aperture* adalah perpustakaan berbasis terbuka untuk *crawler* dan mengindeks sumber informasi seperti sistem berkas, situs web dan kotak surat elektronik. *Aperture* mendukung sejumlah sumber dengan jenis file yang sama dan format dokumen yang tidak umum dan menyediakan cara mudah untuk menjangkaunya dengan implementasi yang terkustomisasi. Adapun komponen dari kode *aperture* adalah *crawling of information sources* seperti *file systems*, *websites*, dan *mail boxes*; *MIME type identification*; *full-text and metadata extraction of various file formats*; dan *opening of crawled resources*.

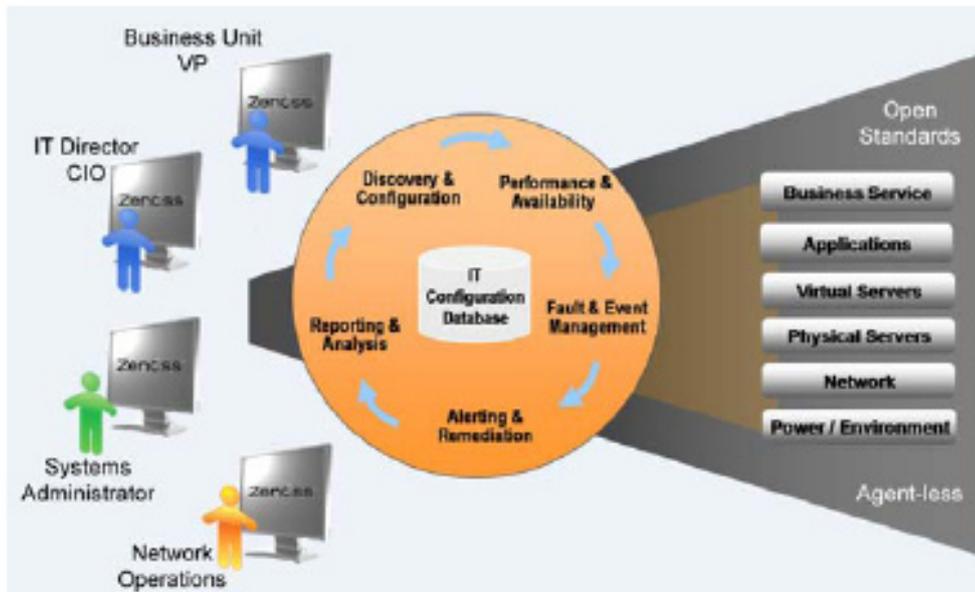
Untuk masing-masing bagian ini, satu set API telah dikembangkan dan sejumlah implementasi disediakan. *Aperture* memiliki fokus yang kuat pada semantik. Sebagai contoh, banyak upaya dilakukan dalam rangka *extractors* untuk dapat juga mengekstraksi sebanyak mungkin metadata yang terdapat dalam sebuah format file (misalnya, judul, penulis, komentar, dan lain lain) dan menggabungkannya dengan sumber metadata spesifik (misalnya lokasi, tanggal modifikasi terakhir, dan lain lain). Semua metadata yang ada dipetakan ke properti dari *namespace NIE* untuk memungkinkan pengolahan secara seragam dari *crawler* dan mengeluarkan informasi. NIE adalah ontologi yang dikembangkan dalam Proyek Nepomuk dan sekarang dikelola oleh komunitas *open source*. *Aperture* bertanggung jawab untuk ekstraksi, tanpa mencoba untuk memberitahu apa yang harus dilakukan dengan data yang terekstraksi tersebut. Anda mungkin ingin menyimpannya, mengindeksnya, atau hanya mengambil teks lengkap dan mencetaknya. Berikut adalah gambaran umum dari proses yang dilakukan perangkat lunak *aperture* (Gambar 2).

Zenoss (Zenoss Core) adalah aplikasi, server dan manajemen jaringan berbasis sumber terbuka berbasis server aplikasi Zope, diterbitkan di bawah lisensi GNU *General Public License* (GPL) versi 2. *Zenoss Core* menyediakan antarmuka berbasis web yang memungkinkan administrator sistem memonitor ketersediaan, *inventory/konfigurasi*, kinerja, dan aktifitas.

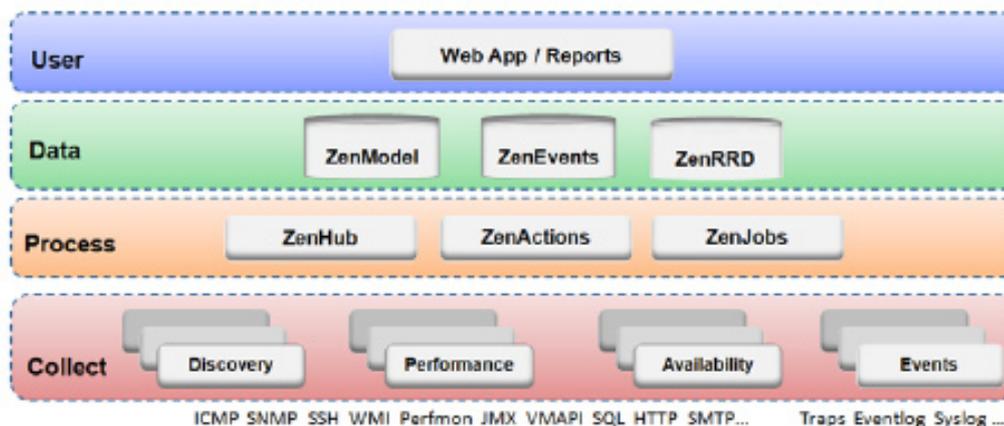
Erik Dahl merintis pengembangan *Zenoss* pada tahun 2002 dan Agustus 2005 mendirikan *Zenoss Inc.*, dengan Bill Karpovich. *Zenoss Inc.* mensponsori pengembangan *Zenoss Core* dan menjual versi enterprise berdasarkan versi *core*. Zope adalah sebuah aplikasi server berbasis web berorientasi objek yang gratis dan sumber terbuka serta ditulis dengan bahasa pemrograman Python. Zope merupakan singkatan dari "*Z Object Publishing Environment*". Aplikasi ini bisa secara penuh diatur memanfaatkan antarmuka pengguna berbasis web (Gambar 3).



Gambar 2 Proses *Aperture* Secara Umum



Gambar 3 Zenoss dari Level Teratas



Gambar 4 Arsitektur Zenoss

Platform

Aplikasi *Zenoss Core* dapat dijalankan pada system-operasi seperti Red Hat Enterprise Linux / CentOS (4, 5), Fedora (Core 9,10), Ubuntu (6.06, 8.04), Debian (5), SuSE (10.X), OpenSUSE (10.3, 11.1), Mac OS X (10.5 Intel, PPC from source), VMWare Appliance (terutama pada Windows), FreeBSD (6.x and 7.x from source), Solaris (10 from source), dan Gentoo (from source).

Zenoss Core diakses via *web-based portal* yang menyediakan akses sistem operasi untuk konfigurasi dan fungsi administrasi. Firefox, dan Internet Explorer didukung sistem ini (Gambar 4).

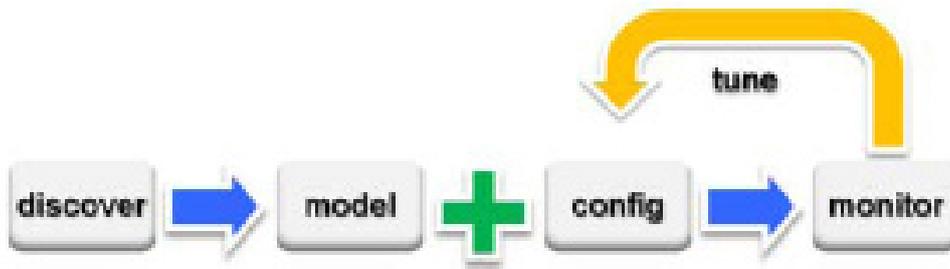
Deskripsi Teknologi

Zenoss Core menggabungkan pemrograman asli dan beberapa proyek *open source* untuk mengintegrasikan *data storage* dan proses *data collection* memanfaatkan *web-based user interface*. *Zenoss Core* dibangun memanfaatkan teknologi *open source* seperti (1) *Zope Application server* yakni web server berorientasi objek, ditulis dalam *Python*; (2) *Python* yakni bahasa pemrograman *extensible*; (3) *Net-SNMP* yakni protokol memonitor yang mengkoleksi status informasi

system; (4) *RRDtool* yakni *graph* dan *log time series data*; (5) *MySQL* yakni *open source database*; (6) *Twisted* yakni sebuah *event-driven networking engine*, ditulis dalam *Python*.

Zenoss Core memberikan manfaat sebagai berikut (1) Memonitor ketersediaan alat jaringan memanfaatkan SNMP, (2) Memonitor layanan jaringan (HTTP, POP3, NNTP, SNMP, FTP), (3) Memonitor sumberdaya *host* (*processor*, *disk usage*) pada sistem informasi dalam jaringan; (4) Memonitor kinerja alat dalam satuan waktu, (5) Extended Microsoft Windows monitoring via Windows Management Instrumentation using SAMBA dan Zenoss open source extensions, (6) Memanfaatkan alat *event management* untuk mengelola *system alerts*, (7) Secara otomatis menemukan sumber daya jaringan dan mengubah konfigurasi jaringan, (8) Sistem kewaspadaan yang menyediakan laporan berbasis set aturan dan kalender tertentu, dan (9) Mendukung format *Nagios plug-in* (Gambar 5 dan Gambar 6).

Secara umum, perbandingan yang dilakukan dalam penelitian terkait kedua perangkat lunak di atas mengacu kepada fungsional yang dimiliki oleh masing-masing dan kemudahan dalam uji coba. Berikut adalah tabel perbandingan keduanya.



Gambar 5 Cara Kerja Zenoss



Gambar 6 Tampilan Zenoss Setelah Instalasi

Tabel 1 Perbandingan Aperture dan Zenoss

No	Item Pemanding	Aperture	Zenoss Core
1	Platform sistem	Java	Windows / Non Windows
2	Bahasa pemrograman	Java	Python
3	Kemampuan menemukan target sistem / file	Ada	Ada
4	Mengelola data	Tidak ada	Ada
5	Menampilkan data	Ada	Ada
6	Plugins	Ada	Ada
7	Penyimpanan data	Ada	Ada
8	WebApp	Ada	Ada
9	Penggunaan script eksternal	Tidak	Ada
10	Menggunakan -exe file	Tidak	Ada

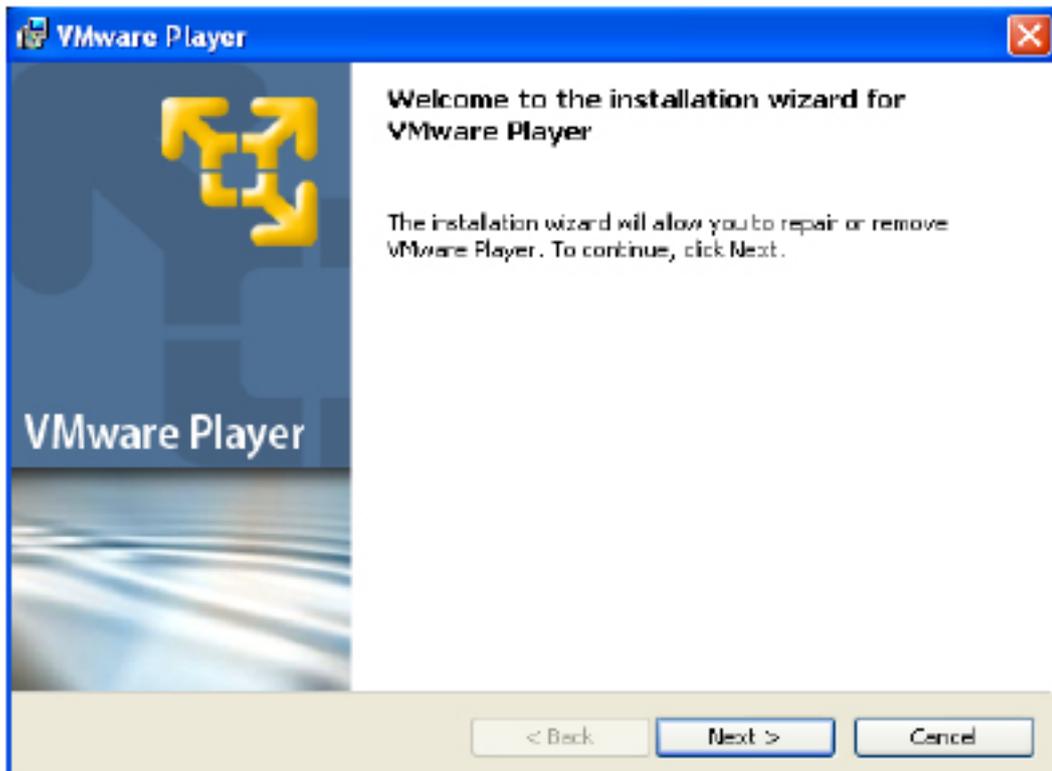
Dari hasil perbandingan di atas, dapat dianalisis bahwa *Zenoss Core* lebih akomodatif terhadap kebutuhan pengguna dalam pemanfaatan aplikasi dengan menyediakan file `-exe` dan bentuk eksternal `script` lainnya untuk mengeksekusi program dalam *platform* yang bervariasi (*windows/non*

windows) sehingga diputuskan bahwa akan dilakukan modifikasi *source code* dan uji coba eksekusi dari *Zenoss Core*. Modifikasi terhadap *source code* yang dilakukan tidak melingkupi semua *script*, tapi lebih kepada menemukan cara agar program yang didesain untuk berjalan di luar *platform windows* dapat dieksekusi. Hal ini ternyata dapat dilakukan dengan memodifikasi program ke dalam konsep mesin virtual sehingga dapat di-load pada *player* mesin virtual.

VMware Player adalah aplikasi desktop gratis yang memungkinkan pengguna untuk melakukan fungsi berikut (1) Menjalankan *virtual machines* pada PC berbasis Windows atau Linux, (2) Menjalankan *virtual machines* tanpa membeli *virtualization software*, menjadikannya mudah memanfaatkan keamanan, fleksibilitas dan portabilitas dari *virtual machines*, (3) Memanfaatkan *host machine devices* seperti pemutar cakram CD dan DVD, dari *virtual machine*, (4) Pada mesin berbasis Windows, VMware Player dapat digunakan untuk membuka dan menjalankan Microsoft Virtual PC dan Microsoft Virtual Server *virtual machines*, juga Symantec Backup Exec System Recovery, formerly LiveState Recovery, *system images*.

Dengan langkah modifikasi tersebut, maka dilaksanakan proses uji coba perangkat dan karena memanfaatkan mesin virtual, maka tidak perlu menginstal program *Zenoss Core*, tapi cukup instalasi Player Mesin Virtual yang dapat dilaksanakan sebagai berikut:

Pertama dilakukan instalasi Player Mesin Virtual (Gambar 7). Lalu, di-load mesin virtual dari *Zenoss Core* (Gambar 8). Bila sudah selesai memuat (Gambar 9) dan sistem siap berjalan, maka akan memberikan tampilan seperti pada Gambar 10.



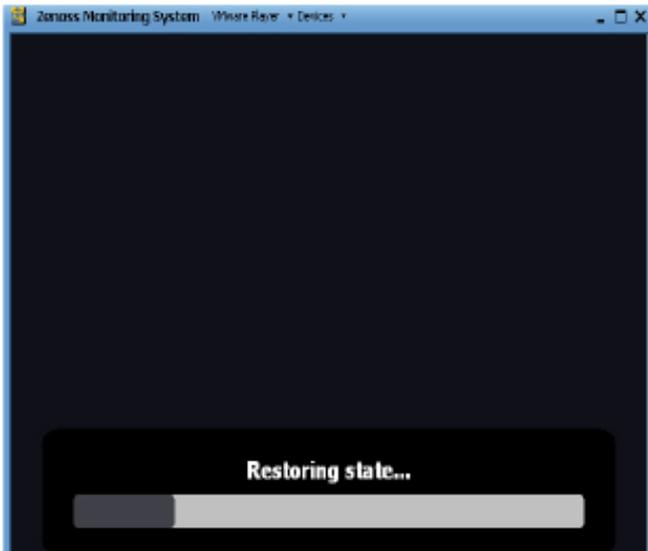
Gambar 7 Proses Instalasi *VMware Player*



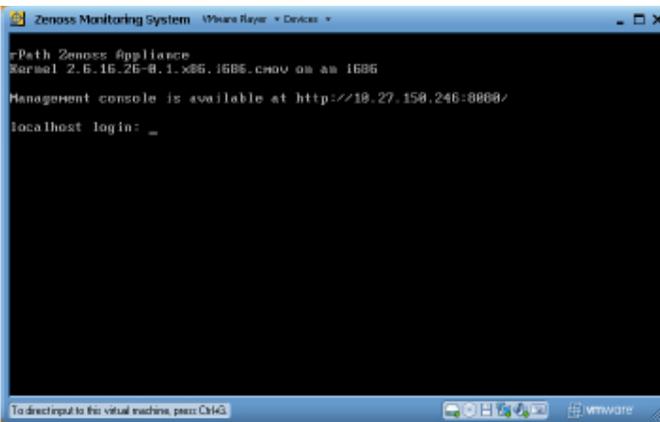
Gambar 8 Memuat Mesin *Virtual Zenoss*

DAFTAR PUSTAKA

- Activity Based Risk Evaluation Model of Auditing, CAATs. Diakses 1 Juli 2008, dari http://www.abrema.net/abrema/caats_g.html.
- Comino, S., and Manenti, F.M. (2003). Open source versus closed source software: Public policies in the software market, Industrial Organization 0306001, EconWPA, revised.
- Potdar, V., and Chang, E. (2004). Open source versus closed source. Proceedings of the 6th International Conference on Enterprise Information Systems, 609-613.



Gambar 9 Zenoss Sedang Dimuat



Gambar 10 Aplikasi Zenoss Siap Digunakan

SIMPULAN

Simpulan yang dapat diambil dari penelitian ini adalah sebagai berikut (1) ZenossCore adalah perangkat lunak berbasis sumber terbuka yang cukup memadai dari fungsional sistem untuk membantu pelaksanaan audit sistem informasi, (2) *Source code* dari ZenossCore dapat direkayasa untuk kebutuhan penggunaan yang lebih personal, dan (3) ZenossCore dapat diimplementasi dan digunakan untuk audit sistem informasi memanfaatkan file *executable* nya. Adapun saran yang dapat diajukan untuk kepentingan penelitian lebih lanjut adalah (1) Sebaiknya perbandingan perangkat lunak lebih diperluas dengan memasukkan perangkat lunak yang sudah mapan dan berbayar, (2) Analisis fungsional dilakukan lebih mendalam satu persatu untuk dilihat aspek kemanfaatan secara riil dilapangan, dan (3) Melakukan uji lapangan untuk perangkat lunak terpilih agar terlihat tingkat kemanfaatan. Skala uji yang disarankan adalah tingkat perguruan tinggi (akademis) dahulu.